

PAT-NO: JP02004206203A
DOCUMENT-IDENTIFIER: JP 2004206203 A
TITLE: DEVICE AND SERVER FOR DISPLAYING DEVICE
INFORMATION
PUBN-DATE: July 22, 2004

INVENTOR-INFORMATION:

NAME	COUNTRY
SASAKUMA, HARUNA	N/A

ASSIGNEE-INFORMATION:

NAME	COUNTRY
MINOLTA CO LTD	N/A

APPL-NO: JP2002371571

APPL-DATE: December 24, 2002

INT-CL (IPC): G06F009/445, G06F012/00

ABSTRACT:

PROBLEM TO BE SOLVED: To enable the display of XML data without an unnecessary increase in storage capacity in a device connected to a network.

SOLUTION: The device connected to one or more client terminals through the network comprises a communication means for performing a communication through the network, a management means for collecting and managing information related to the device, an XML data generation means for generating information related to the device obtained by the management means as XML data, and a data providing mean for providing, in response to a request from a client terminal, the XML data of the information to the client terminal through the communication means.

COPYRIGHT: (C) 2004, JPO&NCIPI

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-206203

(P2004-206203A)

(43) 公開日 平成16年7月22日(2004.7.22)

(51) Int. Cl. 7

F I

テーマコード (参考)

G06F 9/445

G06F 9/06 G10A

5B076

G06F 12/00

G06F 12/00 546Z

5B082

審査請求 未請求 請求項の数 5 O L (全 9 頁)

(21) 出願番号

特開2002-371571(P2002-371571)

(22) 出願日

平成14年12月24日(2002.12.24)

(71) 出願人 000006079

ミノルタ株式会社

大阪府大阪市中央区安土町二丁目3番13

号 大阪国際ビル

(74) 代理人 100062144

弁理士 青山 稔

(74) 代理人 100086405

弁理士 河宮 治

(74) 代理人 100098280

弁理士 石野 正弘

(72) 発明者 笹原 晴宗

大阪府大阪市中央区安土町二丁目3番13

号 大阪国際ビル ミノルタ株式会社内

Fターム(参考) 5B076 AA02

5B082 HA08

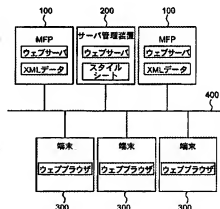
(54) 【発明の名称】 デバイス情報表示のためのデバイスとサーバ

(57) 【要約】

【課題】 ネットワークに接続されたデバイスにおいて、不要に記憶容量を増加することなくXMLデータの表示を可能にする。

【解決手段】 ネットワークを介して1以上のクライアント端末と接続されるデバイスは、ネットワークを介した通信を行う通信手段と、当該デバイスに関する情報を収集し、管理する管理手段と、管理手段により得られるデバイスに関する情報をXMLデータとして生成するXMLデータ生成手段と、クライアント端末からの要求に応じて、前記情報のXMLデータを通信手段を介してクライアント端末に提供するデータ提供手段とを備える。ここで、XMLデータ生成手段は、当該デバイスとは別の外部のサーバ装置に記憶されているスタイルシートを参照するように記述されたXMLデータを生成する。

【選択図】 図1



【特許請求の範囲】

【請求項1】

ネットワークを介して1以上のクライアント端末と接続されるデバイスであって、ネットワークを介した通信を行う通信手段と、当該デバイスに関する情報を収集し、管理する管理手段と、前記管理手段により得られるデバイスに関する情報をXMLデータとして生成するXMLデータ生成手段と、クライアント端末からの要求に応じて、前記情報のXMLデータを前記通信手段を介してクライアント端末に提供するデータ提供手段とを備え、前記のXMLデータ生成手段は、当該デバイスとは別の外部のサーバ装置に記憶されているスタイルシートを参照するように記述されたXMLデータを生成することを特徴とするデバイス。

【請求項2】

前記XMLデータ生成手段は、インターネットを介したファイルの所在によりスタイルシートを参照することを特徴とする請求項1に記載されたデバイス。

【請求項3】

前記デバイスは画像形成装置であることを特徴とする請求項1または2に記載されたデバイス。

【請求項4】

さらに、クライアントの情報に基いて参照すべきスタイルシートを決定するスタイルシート決定手段を備えることを特徴とする請求項1～3のいずれかに記載されたデバイス。

【請求項5】

ネットワークを介して1以上のクライアント端末と接続されるサーバ装置であって、ネットワークに接続される各デバイスが作成する当該デバイスに関する情報のXMLデータを表示するための1以上のスタイルシートを保持するスタイルシート記憶手段と、クライアント端末からの要求に応じてスタイルシート記憶手段に記憶されたスタイルシートを提供するデータ提供手段と、かかるサーバ装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ネットワークを介するXMLデータの情報の表示に関するものである。

【0002】

【従来の技術】

ネットワーク（たとえばLAN）を介して複数の装置が接続されているシステムにおいて、1つの装置が、当該装置についての情報をHTMLデータとして記憶していると、そのHTMLデータを他の装置が受け取って表示（閲覧、印刷など）ができる。HTML（HyperText Markup Language）は、ウェブページを書くために生まれた言語である。たとえば、プリンタなどのデバイスがネットワークを介して利用されるシステムにおいても、当該デバイスが内部の情報（たとえば管理情報）をHTMLデータとして記憶していると、他の装置（たとえばパーソナルコンピュータ端末）はその情報を表示できる。たとえば特開平2001-186301号公報には、HTMLデータを用いた多機能装置（MFP）などのエラー情報の通信が記載されている。

【0003】

HTMLデータの受信には、ページ単位での送信によるネットワークの負荷、データの再利用不可などの問題がある。そこで、XMLを採用して、ウェブページをXMLで作成することが考えられる。XML（eXtensible Markup Language）は、拡張機能付きのマークアップ言語である。XML自体はレイアウトを規定しないので、XMLで記述されたデータ（以下XMLデータという）のレイアウトは、スタイルシート言語（たとえばXSL（eXtensible Stylesheet Language）やCSS（Cascading Style Sheet））で記述したス

イルシートで規定する。また、X S L T (XSL Transformations) は、X S L データの変換処理を行う部分の言語の 1 つである。

【0004】

X M L は種々の用途で利用されるが、データの送信に適している。たとえば、ネットワークに接続されている各種デバイス（プリンタなど）を管理するとき、デバイスの管理情報を X M L データとして保持できる。デバイスの外部の装置のウェブブラウザで管理情報を閲覧するとき、ウェブブラウザは、デバイスに保持されているスタイルシートを参照して X M L データを解析して表示する。

【0005】

【特許文献 1】

特開 2001-186301 号公報

【0006】

【発明が解決しようとする課題】

X M L データの表示を容易にするため、デバイス自体に、X M L をサポートするアプリケーション（たとえばウェブサーバ）を備え、デバイス自体の管理データなどについての X M L データとスタイルシートを記憶しておき、ネットワークを介する外部端末からの要求に応じて X M L データを提供することが考えられる。これによりネットワークへの負荷は軽減される。しかし、この場合、各デバイスにスタイルシートを記憶するための記憶容量が必要である。また、多数のデバイスに同じスタイルシートを記憶している場合、スタイルシートをバージョンアップするとき、デバイスごとにスタイルシートを再インストールする必要がある。また、X M L データは、デバイスやクライアント端末の状況などに応じて、柔軟な表示形態で表示することが望ましいが、多数のスタイルシートを記憶するには大きな記憶容量が必要である。

【0007】

この発明の目的は、ネットワークに接続されたデバイスにおいて、不要に記憶容量を増加することなく X M L データの表示を可能にすることである。

【0008】

【課題を解決するための手段】

本発明に係るデバイスは、ネットワークを介して 1 以上のクライアント端末と接続されるデバイス（たとえば画像形成装置）であって、ネットワークを介した通信を行う通信手段と、当該デバイスに関する情報を収集し、管理する管理手段と、前記管理手段により得られるデバイスに関する情報を X M L データとして生成する X M L データ生成手段と、クライアント端末からの要求に応じて、前記情報の X M L データを前記通信手段を介してクライアント端末に提供するデータ提供手段とを備える。前記の X M L データ生成手段は、当該デバイスとは別の外部のサーバ装置に記憶されているスタイルシートを参照するように記述された X M L データを生成する。

【0009】

このデバイスにおいて、X M L データ生成手段は、たとえば、インターネットを介したファイルの所在によりスタイルシートを参照する。

【0010】

前記のデバイスは、好ましくは、さらに、クライアントの情報に基いて参照すべきスタイルシートを決定するスタイルシート決定手段を備える。

【0011】

本発明に係るサーバ装置は、ネットワークを介して 1 以上のクライアント端末と接続されるサーバ装置であって、ネットワークに接続される各デバイスが作成する当該デバイスに関する情報の X M L データを表示するための 1 以上のスタイルシートを保持するスタイルシート記憶手段と、クライアント端末からの要求に応じてスタイルシート記憶手段に記憶されたスタイルシートを提供するデータ提供手段とからなる。

【0012】

【発明の実施の形態】

以下、添付の図面を参照して本発明の実施の形態を説明する。

図1に示すネットワーク環境では、種々の共用デバイス100、プリンタなど）、種々のサーバ装置200、クライアント端末300（パーソナルコンピュータなど）などがネットワーク（LAN、インターネットなど）400に接続されている。共用デバイス100は、たとえば、複写機、プリンタ、スキャナ、または、それらの機能を複合した多機能装置（MFP）などの画像形成装置を含む。以下では、1例として共用デバイス100がMFPである場合について説明する。クライアント端末300のユーザは、XMLをサポートしているアプリケーション（たとえばウェブブラウザ）を用いて、XMLデータの表示（閲覧、印刷など）をする。XMLデータは、たとえば、デバイスの内部情報であるデバイス管理情報である。具体的には、クライアント端末300のアプリケーション（ウェブブラウザ）は、HTTP（Hypertext Transfer Protocol）によりMFP100のURIを指定してMFP100のウェブサーバにリクエストを送る。ウェブサーバは、クライアント端末からのHTTPによる要求を受けてクライアント端末にデータを提供する。ウェブブラウザは、受け取ったデータを表示する。本実施形態では、ウェブブラウザは、たとえば、XMLデータとスタイルシートを解析して、受け取ったXMLデータをたとえばHTMLデータに変換して、表示（閲覧、印刷など）を行う。

【0013】

ここで、XMLデータを表現するための1以上のスタイルシートは、MFP100とは別のサーバ装置200に記憶しておく。なお、サーバ装置200は、同じLAN内に設けてもよいし、LAN外の、たとえばインターネット経由でアクセスするサーバであってもよい。XMLデータの文書型宣言部は、そのスタイルシートを指定する記述を含んでいる。前述のデバイス管理情報のXMLデータの場合、サーバ200に記憶されているスタイルシートを指定する。クライアント端末300のウェブブラウザは、MFP100からXMLデータを受け取ると、XMLデータに含まれる記述に従ってサーバ装置200に記憶されているスタイルシートを参照してXMLデータの表示をする。

【0014】

スタイルシートをデバイス100とは別のサーバ装置200に記憶しておくので、多種多様なスタイルシートを記憶できる。たとえば、ユーザごとにカスタマイズした任意のスタイルシートを記憶できる。カスタマイズにより、デバイス、クライアントなどの要望に応じた柔軟な表示レイアウトが可能になる。また、日本語、英語などの複数の異なる言語のスタイルシートを記憶できる。また、障害者を考慮したユニバーサルデザインを用いたスタイルシートも記憶できる。

【0015】

スタイルシートをデバイス100とは別のサーバ装置200に記憶しておくことにより種々の有用な効果が得られる。複数のスタイルシートをデバイス100とは別のサーバ装置200に記憶しておくことにより、デバイス100には、スタイルシートを格納するためのメモリの記憶容量が不要になる。また、同じスタイルシートを多数のデバイス100（たとえば同じ機能のプリンタ）に記憶させている場合、スタイルシートのバージョンアップはデバイスごとに行う必要がある。これに対し、サーバ装置200内の1つの同じスタイルシートを参照させるようにすると、サーバ装置200のそのスタイルシートをバージョンアップするだけで対処できる。また、記憶容量の大きい記憶装置に、多数のスタイルシートを記憶しておき、その中から選択することにより、XMLデータを多様なレイアウトで表示できる。たとえば、ユーザごとにスタイルシートを用意しておく、ユーザごとに任意のカスタマイズしたスタイルシートとすることができ、また、日本語用スタイルシート、英語用スタイルシートなど、各言語用スタイルシートを用意しておけばデバイスの設置場所に対応した複数の異なる言語（日本語、英語など）の中からの選択も容易になる。また、障害者を考慮したユニバーサルデザインを用いたスタイルシートの指定も容易になる。

【0016】

スタイルシートが複数用意されている場合は、MFP設置時に1つのスタイルシートに決

めておいてもよいし、MFPが種々の条件に応じて決定するようにしてもよい。たとえば、MFPがアクセスしてきたクライアント端末のIPアドレスやユーザ名を参照して決定するようにしてもよい。たとえば、IPアドレスからどの国にあるクライアント端末であるかを判断し、その国に対応した言語のスタイルシートを選択する。また、あらかじめユーザまたはIPアドレスごとにスタイルシートが関連付けられているのであれば、対応するスタイルシートを選択する。そして、選択されたスタイルシートを参照するように記述されたXMLデータを生成する。

【0017】

次に、ネットワーク400に接続される各装置について具体的に説明する。共用デバイス100の1つである多機能装置(MFP)は、プリンタ、スキャナなどとして機能するデバイスである。多機能装置は、たとえば、図2に示すように、原稿の画像を読み取るスキャナ装置102、印刷データを受け取って紙に印刷するプリントエンジン部104、および、それらを制御するコントローラ106を備える。コントローラ106は、操作パネル部108、外部のネットワーク400に接続するネットワークインタフェースを含む通信装置110などと接続される。操作パネル108は、使用者による各種入力設定や表示を行う。プリンタとして機能する場合、ネットワークを介した通信を行う通信装置110を介して外部から印刷データを受け取ると、ラスタデータを作成し、プリントエンジン部104を制御して印刷を行う。また、コントローラ106は、操作パネル108を介する設定や各種センサ(図示しない)の出力結果を基にデバイス自身の管理データを作成し、要求があると提供する。

【0018】

コントローラ106は、全体を制御するCPU120、記憶装置(HDD)122、画像処理部124、画像メモリ126などを備える。記憶装置122は、たとえばハードディスク装置であり、多機能装置を制御する制御プログラム130の他に、当該多機能装置に関する情報を収集し管理する状態管理部132と、XMLデータ生成部134と、ウェブサーバ136とを記憶する。また、生成されたXMLデータを記憶するXMLデータ記憶部138を備える。XMLデータ生成部134は、状態管理部132が収集した当該多機能装置に関する情報のXMLデータを生成する。なお、このXMLデータは、当該デバイスとは別の外部のサーバ装置に記憶されているスタイルシートを参照するように記述される。ウェブサーバ136は、外部のクライアントから管理データの閲覧要求があると、XML生成部134により生成されたXMLデータを通信装置110を介して提供する。

【0019】

サーバ装置200は、MFP100とは別にネットワークに接続されている共用装置であり、図3に示すように、通常のコンピュータと同様に、全体を制御するCPU202、キーボードなどの入力装置204、ディスプレイ装置206、記憶装置(HDD)208及び、ネットワーク500に接続するネットワークインタフェースを含む通信装置210を備える。記憶装置208には、OSプログラム(図示しない)の他、ウェブサーバ220と1以上のスタイルシート222などを記憶する。

【0020】

クライアント端末300は、たとえばパーソナルコンピュータ(PC)であり、パーソナルコンピュータは、たとえば、図4に示すように、全体を制御するCPU302、キーボードなどの入力装置304、ディスプレイ装置306、記憶装置(HDD)308及び、ネットワーク400に接続するネットワークインタフェースを含む通信装置310を備える。記憶装置308は、OSプログラム(図示しない)の他、XMLデータを表示するためのアプリケーションであるウェブブラウザ320を格納している。

【0021】

このネットワーク環境において、使用者は、クライアント端末300でウェブブラウザ320を起動する。そのウェブブラウザ320で、MFP100のURIを入力することにより、MFP100のウェブサーバ136と接続を行う。

【0022】

MFP100において、ウェブサーバ136は、クライアント端末300のウェブブラウザ320から管理データ要求を受け取ると、XMLデータ生成部134に管理データを要求する。XMLデータ生成部134は、状態管理部132からMFP100の管理データ情報を取得し、そのXMLデータを作成する。ウェブサーバ136は、作成されたXMLデータをクライアント端末300に送信する。

【0023】

図5は、管理データのXMLデータの1例である。この管理データによれば、以下のことが分かる。このMFPの製品名は「abc-001」である。原稿トレイは装着されており、用紙トレイは装着されており、その名称はトレイ1であり、用紙サイズはレターであり、媒体は普通紙であり、用紙の搬送方向は短辺であり、トレイの用紙収容容量は250枚であり、用紙はトレイに収納されている。このXMLデータの2行目にスタイルシートが指定されている。ここで、href=の後にスタイルシートのURIが記述され、type=の後にスタイルシートのタイプが記述される。

【0024】

クライアント端末300のブラウザ320は、XMLデータを受け取ると、XMLデータに指定されたスタイルシートを要求する。この例では、サーバ装置200に格納されているスタイルシートが指定されているので、サーバ装置200のウェブサーバ220にスタイルシートを要求する。これに応じて、ウェブサーバ220は、要求されたスタイルシートをクライアント端末300に送信する。図6は、図5のXMLデータで指定されたスタイルシートの1例である。スタイルシートを受け取ったクライアント端末300のブラウザ320は、MFP100から受け取ったXMLデータとサーバ装置200から受け取ったスタイルシートとを解析し、たとえばHTMLデータに変換して画面に表示する。

【0025】

図7は、クライアント端末300のフローチャートである。ユーザによりウェブブラウザ320が起動されたとき、MFP100のウェブサーバ136のURIが指定されていると(S100でYES)、ウェブサーバ136にXMLデータの要求を送信する(S102)。なお、ウェブサーバ136のURIを指定していない場合は(S100でNO)、他の処理をする(S104)。

【0026】

MFP100からXMLデータを受信すると(S106でYES)、受け取ったXMLデータにスタイルシートの指定がされているとき(S110でYES)クライアントの指定に基づいて参照すべきスタイルシートを決定し、指定されたサーバ装置200にスタイルシートの要求を送信する(S112)。MFP100からXMLデータが受信されていない場合は(S106でNO)、ウェブブラウザ320の所定の設定時間を経過していない場合は(S108でNO)、ステップS106に戻ってXMLデータの送信をチェックする。設定時間は、XMLデータの送信に必要な時間を考慮して設定する。設定時間を経過すると(S108でYES)、XMLデータ受信の処理を打ち切って終了する。

【0027】

MFP100からのXMLデータが受信されている場合に(S106でYES)、次に、サーバ装置200からスタイルシートが受信されると(S114でYES)、XMLデータとスタイルシートを解析し、スタイルシートの指定に従ってデータを表示するHTMLデータを生成する(S118)。これによりHTMLデータの表示が行える。ウェブブラウザ320の所定の設定時間を経過していない場合は(S116でNO)、ステップS114に戻ってスタイルシートの送信をチェックする。設定時間は、スタイルシートの送信に必要な時間を考慮して設定する。ウェブブラウザ320の設定時間を経過すると(S116でYES)、スタイルシート受信の処理を打ち切って終了する。

【0028】

図8は、MFP100のXMLデータ生成部134のプログラムのフローチャートである。クライアント端末(PC端末)300からMFP100のウェブサーバ122にデバイス情報の要求があると、状態管理部132よりデバイス情報を取得して(S200でYES

S)、取得したデバイス情報をもとにXMLデータを作成する(S202)。XMLデータを作成すると(S204でYES)、ウェブサーバ120に提供してXMLデータを送らせる(S206)。これにより、クライアント端末300でXMLデータの表示が行える。なお、このMFP100で実行するプログラムは、当業者に知られているように、スクリプト、外部プログラムなどの種々の形態で実装できる。

【0029】

図9は、サーバ装置200のウェブサーバ220のフローチャートである。クライアント端末(PC端末)300からスタイルシートの要求があると(S300でYES)、その指定されたスタイルシートが存在すれば(S302でYES)、そのスタイルシートをクライアント端末300に送信する(S304)。指定されたスタイルシートが存在しなければ(S302でNO)、処理を終了する。 10

【0030】

なお、前述の実施の形態では、XMLデータを提供するデータ提供手段としてウェブサーバを用い、XML閲覧手段としてウェブブラウザを用いた。しかし、前述の実施形態に必要な機能を備える他のアプリケーションを用いてもよい。

【0031】

【発明の効果】

XMLデータを格納するデバイスとは別のサーバ装置にスタイルシートを置くことにより、デバイスにはスタイルシートを格納するための記憶容量が必要でない。

また、複数の装置で使用される1つの同じスタイルシートのバージョンアップが、サーバ装置のスタイルシートのバージョンアップをするのみで容易に対処できる。 20

【図面の簡単な説明】

【図1】第1の実施の形態に係るネットワークシステムのブロック図

【図2】MFPのブロック図

【図3】サーバのブロック図

【図4】クライアント端末のブロック図

【図5】デバイス管理装置のXMLファイルの1例の図

【図6】スタイルシートの1例の図

【図7】クライアント端末のフローチャート

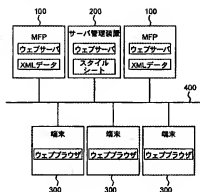
【図8】MFPのXMLデータ生成部のフローチャート 30

【図9】サーバ装置のウェブサーバのフローチャート

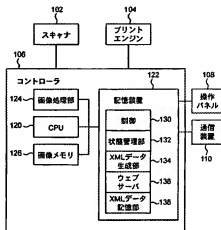
【符号の説明】

100 デバイス(たとえばMFP)、 102 通信装置、 106 コントローラ、 112 CPU、 114 記憶装置、 122 状態管理部、 124 ウェブサーバ、 126 XML生成プログラム、 200 サーバ装置、 202 CPU、 208 記憶装置、 210 通信装置、 220 ウェブサーバ、 222 スタイルシート記憶部、 300 クライアント端末、 302 CPU、 320 ウェブブラウザ。

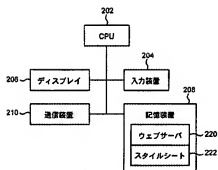
【図 1】



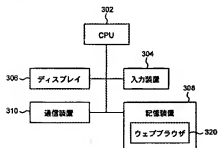
【図 2】



【図 3】



【図 4】



【図 5】

```

<xml version="1.0" encoding="Shift_JIS">
<xml-stylesheet href="http://www.minolta.co.jp/trustidea.xml" type="text/xml">
<MFP ProductName="abc-011">
<Summary>
<InputTray Installed="YES">
<Tray Installed="YES">
<Name>Tray1</Name>
<PaperSize>Letter</PaperSize>
<MediaType>Plain</MediaType>
<Orientation>Short Edge</Orientation>
<Capacity>250</Capacity>
<Paper>Ready</Paper>
</Tray>
</InputTray>
</Summary>
</MFP>

```

```
<html version='1.0' encoding='Shift_JIS'>  
  <body><meta charset='http://www.w3.org/1997/xsltform version=1.0' xmlns:xsl=  
    h=http://www.w3.org/1998/xslt?>  
    <xsl:output method='html' encoding='Shift_JIS'/>  
  
    <xsl:template match='/BODY'/>  
  
      <HTML>  
  
        <HEAD>  
  
          <HEAD>  
  
            <BODY>  
  
              <TABLE>  
  
                <TR>  
  
                  <TD>Mixed Original</TD>  
  
                  <TD><xsl:value-of select='SummaryPageInfo/Freq/Pagesize'/></TD>  
  
                </TR>  
  
                <TR>  
  
                  <TD>Split Original</TD>  
  
                  <TD><xsl:value-of select='SummaryPageInfo/Freq/Pagesize'/></TD>  
  
                </TR>  
  
              </TABLE>  
  
            <BODY/>  
  
          </HTML>  
  
        </xsl:template>  
  
      </body></html>
```

```

graph TD
    S100[S100 MFPGのウェブサーバのURLを指定したか?] -- NO --> S104[S104 他の処理]
    S100 -- YES --> S102[S102 MFPGにXMLデータの要求を送信する]
    S102 --> S106[S106 MFPGよりXMLデータが返信されてきたか?]
    S106 -- NO --> S104
    S106 -- YES --> S110[S110 受け取ったXMLデータにスタイルシートが指定されているか?]
    S110 -- YES --> S112[S112 指定されたサーバにスタイルシートの要求を送信する]
    S112 --> S114[S114 サーバよりスタイルシートが返信されてきたか?]
    S114 -- YES --> S118[S118 XMLデータとスタイルシートを解析しHTMLデータを生成する]
    S114 -- NO --> S116[S116 ブラウザの設定値以上の頻度が経過したか?]
    S116 -- YES --> S118
    S116 -- NO --> S104
    S103[S103 ブラウザの設定値以上の頻度が経過したか?] -- NO --> S104
    S103 -- YES --> S106
  
```

```

graph TD
    S200([クライアントPCから  
XMLデータの要求]) --> S202{状態管理部より  
デバイス情報を取得  
できたか?}
    S202 -- NO --> S204{取得したデバイス  
情報をもとにXMLを作成  
できたか?}
    S202 -- YES --> S202[XMLデータ生成]
    S204 -- NO --> S206[XMLデータをもPC端末に送信]
    S204 -- YES --> S206
    S206 --> End([終了])
  
```

```

graph TD
    Start([スタート]) --> S300{S300  
PC端末から  
スタイルシートの要求  
があるか?}
    S300 -- NO --> S301([S301  
ウェブサーバから  
スタイルシートを  
取得する])
    S301 --> S302{S302  
指定された  
スタイルシート  
があるか?}
    S302 -- NO --> S303([S303  
ウェブサーバから  
スタイルシートを  
取得する])
    S303 --> S302
    S302 -- YES --> S304[S304  
スタイルシートをPC端末に送信]
    S304 --> End([終了])
  
```